PATENT COOPERATION TREATY

From the
INTERNATIONAL SEARCHING AUTHORITY

<table>
<tr><td>To:

see form PCT/ISA/220</td><td>

PCT

WRITTEN OPINION OF THE INTERNATIONAL SEARCHING AUTHORITY
(PCT Rule 43*bis*.1)

</td></tr>
</table>

Date of mailing *(day/month/year)*   see form PCT/ISA/210 (second sheet)

Applicant's or agent's file reference see form PCT/ISA/220	**FOR FURTHER ACTION** See paragraph 2 below

International application No. PCT/IB2004/003549	International filing date *(day/month/year)* 28.09.2004	Priority date *(day/month/year)* 30.09.2003

International Patent Classification (IPC) or both national classification and IPC G06F17/24, G06F17/30

Applicant CANON KABUSHIKI KAISHA

1. This opinion contains indications relating to the following items:

    ☒ Box No. I   Basis of the opinion

    ☐ Box No. II   Priority

    ☐ Box No. III   Non-establishment of opinion with regard to novelty, inventive step and industrial applicability

    ☐ Box No. IV   Lack of unity of invention

    ☒ Box No. V   Reasoned statement under Rule 43*bis*.1(a)(i) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

    ☐ Box No. VI   Certain documents cited

    ☐ Box No. VII   Certain defects in the international application

    ☐ Box No. VIII   Certain observations on the international application

2. **FURTHER ACTION**

    If a demand for international preliminary examination is made, this opinion will usually be considered to be a written opinion of the International Preliminary Examining Authority ("IPEA"). However, this does not apply where the applicant chooses an Authority other than this one to be the IPEA and the chosen IPEA has notifed the International Bureau under Rule 66.1*bis*(b) that written opinions of this International Searching Authority will not be so considered.

    If this opinion is, as provided above, considered to be a written opinion of the IPEA, the applicant is invited to submit to the IPEA a written reply together, where appropriate, with amendments, before the expiration of three months from the date of mailing of Form PCT/ISA/220 or before the expiration of 22 months from the priority date, whichever expires later.

    For further options, see Form PCT/ISA/220.

3. For further details, see notes to Form PCT/ISA/220.

Name and mailing address of the ISA: European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Authorized Officer Michalski, S Telephone No. +49 89 2399-6977

IAP2C   23 JAN 2006

Box No. I   Basis of the opinion

1. With regard to the **language**, this opinion has been established on the basis of the international application in the language in which it was filed, unless otherwise indicated under this item.

    ☐   This opinion has been established on the basis of a translation from the original language into the following language   , which is the language of a translation furnished for the purposes of international search (under Rules 12.3 and 23.1(b)).

2. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application and necessary to the claimed invention, this opinion has been established on the basis of:

    a. type of material:

        ☐   a sequence listing

        ☐   table(s) related to the sequence listing

    b. format of material:

        ☐   in written format

        ☐   in computer readable form

    c. time of filing/furnishing:

        ☐   contained in the international application as filed.

        ☐   filed together with the international application in computer readable form.

        ☐   furnished subsequently to this Authority for the purposes of search.

3. ☐   In addition, in the case that more than one version or copy of a sequence listing and/or table relating thereto has been filed or furnished, the required statements that the information in the subsequent or additional copies is identical to that in the application as filed or does not go beyond the application as filed, as appropriate, were furnished.

4. Additional comments:

Box No. V   Reasoned statement under Rule 43bis.1(a)(i) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

---

1. Statement

   Novelty (N)   Yes: Claims   1-25
                                      No: Claims

   Inventive step (IS)   Yes: Claims
                                      No: Claims   1-25

   Industrial applicability (IA)   Yes: Claims   1-25
                                      No: Claims

2. Citations and explanations

   see separate sheet

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING
AUTHORITY (SEPARATE SHEET)

10/565501
International application No.

23 JAN 2006

PCT/IB2004/003549

Re Item V
Reasoned statement with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1   The following documents are referred to in this communication:

-D1 : PRAMOD JAIN: "Draw and annotate in your browser using SVG" INTERNET, [Online] 8 May 2002 (2002-05-08), pages 1-6, Retrieved from the Internet: URL:http://builder.com.com/5100-6371-10449 78.html> [retrieved on 2004-05-19]

-D2 : FORBES D: "Build Flexible, Lightweight XML-Based Images for ASP.NET Using Scalable Vector Graphics" MSDN MAGAZINE, [Online] July 2003 (2003-07), pages 1-8, Retrieved from the Internet: URL:http://msdn.microsoft.com/msdnmag/issues/03/07/ScalableVectorGraphics/default.as   px> [retrieved on 2004-05-19]

-D3 : WENZ C, HAUSER T: "Scripting SVG" SVG OPEN, CARTO.NET DEVELOPERS CONFERENCE, [Online] 17 July 2002 (2002-07-17), pages 1-7, ZURICH, SWITZERLAND Retrieved from the Internet: URL:http://www.svgopen.org/2002/papers/hauser_wenz__scripting_svg/index.html> [retrieved on 2004-05-19]

2   INDEPENDENT CLAIMS

2.1   The present application does not meet the criteria of Article 33(1) PCT, because the subject-matter of claim 1 does not involve an inventive step in the sense of Article 33(3) PCT. Document D1 discloses (the references in parentheses applying to this document):

a method of interacting via a web browser with a digital graphical document ("circle.svg") represented in the SVG language (see paragraph 1). The method of D1 comprises:

i) receiving the original document ("circle.svg") in read mode (implicit in D1, listing B; the original document being retrieved from the web browser), transforming said original document into an editable version (implicit in D1, since D1 mentions in page 1 "how mouse events can be captured using the SVG api and how the DOM can be changed") according to a set of predefined transformation rules (see listing B, "drawLine()" function) incorporating a set of rules for writing to the document (see listing B, "SVGDoc.createElement()"

ii) interacting via the browser in order to modify the editable version according to the set of writing rules (see listing C, "onmouseup="OMU(evt)"", and the call to "drawLine" in the "OMU()" function)

The subject-matter of claim 1 therefore differs from this known method in that claim 1 specifies a further step of transforming the modified version into a version in read mode.

The problem to be solved in claim 1 may therefore be regarded as how to implement a tool for editing a graphical document in a web browser.

When facing this problem, the skilled person would clearly contemplate using the teaching of D1, since D1 aims at "treating a browser like a drawing or painting tool" using SVG documents (see D1, paragraph 1). Since D1 mentions the possibility of "drawing and saving" (see D1, paragraph 1), it would be obvious for the skilled person to decide saving the modified document, by e.g. transmitting the content of the modified document to the web server of said document, and storing thereon said modified version for other read requests.

Hence, the subject matter of claim 1 can not be considered as involving an inventive step (Article 33(3)) PCT.

2.2 It should also be noted that the subject matter of claim 1 can not be considered as involving an inventive step in the light of D2, which discloses a method of modifying a SVG document ("the existing content can be completely modified") using a browser interface (see D2, figure 5; page 2, last paragraph - page 3, first paragraph) and from which the implementation of step (iii) of claim 1 would be obvious.

2.3 It is also observed that several *browser based* editors for XML or HTML (e.g. the "WebEditor Control" of SJ Namo Interactive, an ActiveX control for modifying HTML files) are known at the date of priority of the present application. Even if it may be argued that a HTML document is not a graphical one, deciding to implement a browser based editor for modifying specifically graphical documents can not be considered as involving an inventive step (Article 33(3) PCT), since using a browser based editor is known and since the benefits of using such an editor for graphical documents are highly predictable.

2.4 The subject matters of independent claims 13, 24 and 25 correspond in terms of an apparatus (respectively: a medium and a program) to the subject matter of independent claim 1 and can not, in view of the above, be considered as involving an inventive step (Article 33(3) PCT).

3   DEPENDENT CLAIMS 2-12, 14-23

Dependent claims 2-12, 14-23 do not contain any features which, in combination with the features of any claim to which they refer, meet the requirements of the PCT in respect of inventive step (Article 33(3) PCT) with regard to the disclosure of D1, the reasons being as follows:

-Claims 2, 14 : using transforming rules not related to the content of the document to modify is an obvious requirement in any editor system. This feature may also be considered as employed in listing B of D1, since the DrawLine function is not related to the specific content of the SVG file being modified.

-Claim 3 : obvious additional feature in a browser based editor.

-Claims 4, 15 : cancelling modifications is common place in any editor system:

-Claims 5, 6, 16, 17 :

   -document D1 teaches realising a SVG browser based editor using script embedded in HTML code (see listing B and C) and SVG code (see listing C, callbacks to onMouseXXX() events), added to the actual SVG content to modify. Since it is clear that the aim of the editor suggested by D1 is the modification of the *original* document, it would be obvious to the skilled person to decide helping to remove superfluous scripting code when saving the document. Known ways to do so is to identify code blocks using block delimiters or setting attributes for blocks.

   -It should also be noted that the browser based editor suggested by D1 has inherently to be provided with instructions for saving the modified document on a web server. One straightforward possibility for achieving this goal would be to store the web server address in a variable during step (i), for subsequent use during the step of storing the modified document.

   -It is also observed that the scripts required for realising a browser based SVG editor can be embedded in HTML and/or in the SVG document itself, as pointed out in document D3, page 5 ("apart from putting the ECMAScript code in the SVG file itself,..."). In order to be able to store a modified content free from superfluous scripts, it would be a straightforward measure to help identifying said scripts for removal before the concrete storing.

   -It should also be noted that the scripts required for realising a browser based SVG editor may obviously be embedded during an XSL transformation and removed after modification of the SVG document using a reverse XSL transformation. Initialising script tags with attributes during the first transformation in order to help the reverse transformation is an obvious possibility for the skilled programmer.

-Claims 7, 18 : identifying elements to be modified in a structured document is

obvious. In D3, figure 4, the "keytext" element of a SVG document is specifically identified for later modifications corresponding to keyboard actions. Identifying selectable elements for keyboard or mouse interactions is obvious in a graphical user interface of an editor.

-Claims 8, 9, 19, 20 : deactivating animations is obvious when editing an animated document, such as a SVG document. Moving code portions of nodes corresponding to animations into a portion not handled by the display software, and restoring said portions upon storing is a straightforward design measure for doing so in the context of a browser based editor. Deciding to deactivate animations based on a parameter is merely an implementation detail.

-Claims 10, 21 : insofar as these claims can be understood, deciding to add a "modify" button in order to enter a modification step, and deciding to handle the corresponding logic, are obvious in any document editor.

-Claims 11, 22 : the browser based editing method of D1 can be construed as a distributed architecture, wherein the document is edited remotely from its original location.

-Claims 12, 23 : storing variables modified during a modification step (such as keyboard input) is obvious in any editor. Storing said modified variables is merely an implementation detail in an editor for SVG content.